|  |
| --- |
|  |

Lesson 12: The Graph of the Equation

Student Outcomes

* Students understand the meaning of the graph of , namely .
* Students understand the definitions of when a function is increasing or decreasing.

Lesson Notes

This lesson is designed to make sense of the definition of the *graph of the equation ,* which is quite different from the definition of the *graph of* covered in the previous lesson. The ultimate goal is to show that the “graph of ” and the “graph of ” both define the same set in the Cartesian plane. The argument that shows that these two sets are the same uses an idea that is very similar to the “vertical line test.”

Lesson 12 also addresses directly three concepts that are usually “swept under the rug” in K–12 mathematics and in ignoring, can create confusion about algebra in students’ minds. The first two concepts are described in the Lesson Notes of Lesson 11. This lesson uses pseudo code to build a concept image of a solution set that is different from the “sifting” and “gateway” concept images used in Module 1. In doing so, we once again show how number sentences play an important role in testing for solutions in the solution set to an equation—that equations are always about numbers, not about letters.

In the last lesson, the graph of was created by a generative process (using a for-next loop). In this lesson, students will see that the graph of is created by an “if-then” testing process.

**Note about lesson pace:** Just like the last lesson, use the fact that the examples are very similar to reduce the amount of “writing on the board” you will have to do. If you use computer projection, point out the similarities to your students.

Classwork

Classwork

In Module 1, you graphed equations such as by plotting the points on the Cartesian coordinate plane that corresponded to all of the ordered pairs of numbers that were in the solution set. We called the geometric figure that resulted from plotting those points in the plane the “graph of the equation in two variables.”

In this lesson, we extend this notion of the graph of an equation to the graph of for a function . In doing so, we use computer “thought code” to describe the process of generating the ordered pairs in the graph of .

If necessary, recall for your students the steps to describe the solution set of an equation using set notation from Module 1. For example, to describe the set of solutions to the equation , we use

where “{” means “the set of” and the vertical bar “|” means “such that.” Therefore, the set notation reads, “The set of all real numbers such that is true.” In this case, this set is , which we imagined was created by stepping through all real numbers and finding solutions using the following “sifting” or “gateway” procedure: Substitute each real number in for into the equation; if the resulting number sentence is true for that number, include the number in the solution set; otherwise discard it.

Example 1 (4 minutes)

Example 1

In the previous lesson, we studied a simple type of instruction that computers perform called a for-next loop. Another simple type of instruction is an *if-then statement*. Below is example code of a program that tests for and prints “True” when ; otherwise it prints “False.”

|  |
| --- |
| **Declare integer For all from 1 to 4  If then  Print True  else  Print False  Endif Next** |

The output of this program code is:

False  
True  
False  
False

Ask your students to interpret the output: can you describe how to relate it back to the code? Then go through the code step-by-step with your students as if you and the class were a “computer.”

Notice that the if-then statement in the code above is really just testing whether each number in the loop is in the solution set!

Example 2 (4 minutes)

Before going through the code step-by-step with your students, ask your students to “translate the code into English.” For example, ask what the effect the declaration of as an integer has on the “for” statement? (It limits to the numbers 0, 1, 2, 3, 4.) In general, is it possible that the set could be empty after the for-next loop? (Yes.)

Example 2

Perform the instructions in the following programming code as if *you were a computer and your paper was the computer screen*:

|  |
| --- |
| **Declare integer Initialize as {} For all from 0 to 4  If then  Append to   else  Do NOT append to   Endif Next  Print** |

Output:

Go through the code step-by-step with your students as if you and the class were a “computer.”

Discussion (2 minutes)

Discussion

Compare the for-next/if-then code above to the following set-builder notation we used to describe solution sets in Module 1:

Check to see that the set-builder notation also generates the set . *Whenever you see set-builder notation to describe a set, a powerful way to interpret that notation is to think of the set as being generated by a “program” like the for-next/if-then code above.*

Point out all of the similarities between the set-builder notation and the pseudo code above. For example, where is the variable specified as an integer in both? During each step of the loop, is an actual number (, then , then , …); hence, it is possible to determine whether the equation is true or false. Similarly, what does it mean for and to both be true? (We will replace “ integer” later in the lesson with , replace “” with , and ” with )

Exercise 1 (10 minutes)

Exercise 1

Next we write code that generates a graph of a *two variable equation* for in and in . The solution set of this equation is generated by testing each ordered pair in the set,

to see if it is a solution to the equation. Then the graph is just the plot of solutions in the Cartesian plane. We can instruct a computer to find these points and plot them using the following program:

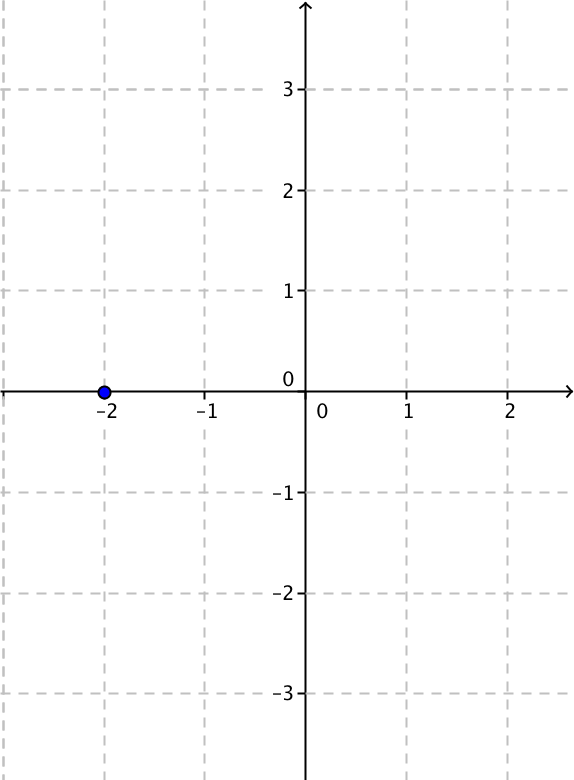
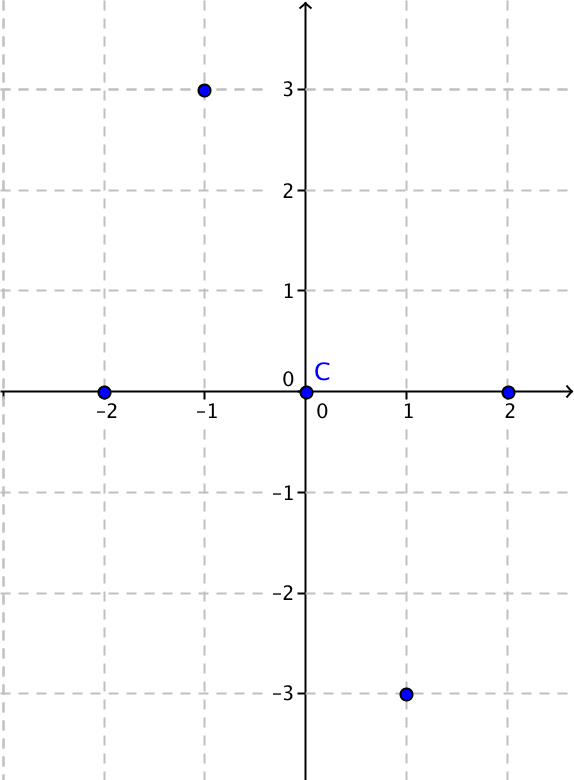
|  |  |  |
| --- | --- | --- |
| **Declare and integers Initialize as {} For all in   For all in   If then  Append to   else  Do NOT append to   Endif  Next  Next  Print  Plot** |  | **Loops through each for Tests whether , then for , is a solution. and so on (see arrows in  table below).** |

* 1. Use the table below to record the decisions a computer would make when following the program instructions above. Fill in each cell with “Yes” or “No” depending on whether the ordered pair would be appended or not. (The step where has been done for you.)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |
|  | No | Yes | No | No | No |
|  | Yes | No | Yes | No | Yes |
|  | No | No | No | Yes | No |

Help students understand the nested for-next loop: First, is kept fixed while loops through , , and . The “Next ” command then fixes while again steps through 3, 0, and 3. This continues until and . Arrows are drawn on your table to indicate the order of this process.

* 1. What would be the output to the “Print ” command? (The first ordered pair is listed for you.)  
       
     Output:  
     { , \_\_\_\_\_\_\_\_\_\_, \_\_\_\_\_\_\_\_\_\_, \_\_\_\_\_\_\_\_\_\_, \_\_\_\_\_\_\_\_\_\_ }
  2. Plot the solution set in the Cartesian plane. (The first ordered pair in has been plotted for you.)

Solution

Point out that the “Yes” answers in the table show a similar pattern as the students’ graphs. The point that you want to make is that this code checks *every* ordered pair in the domain and range and appends only those to the set that solve the equation. In the next exercise, we will see that this is the same as set-builder notation.

Exercise 2 (15 minutes)

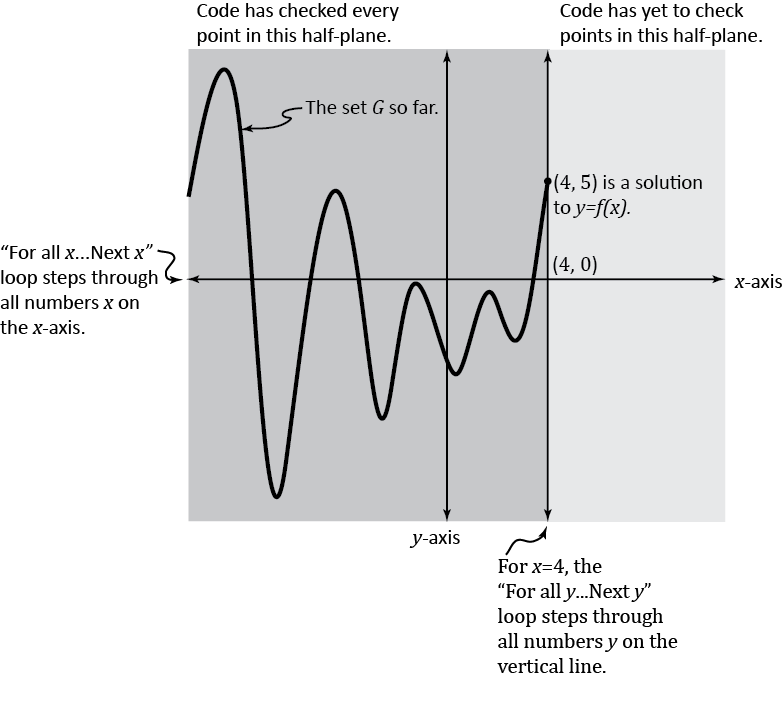
Exercise 2

The program code in Exercise 3 is a way to imagine how set-builder notation generates solution sets and figures in the plane. Given a function with domain and range all real numbers, a slight modification of the program code above can be used to generate the graph of the equation :

Even though the code below cannot be run on a computer, we can run the following “thought code” in our minds:

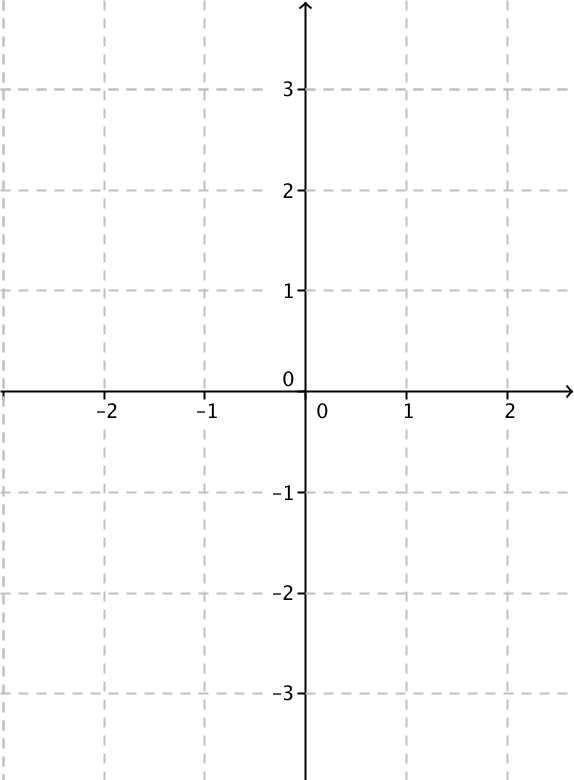
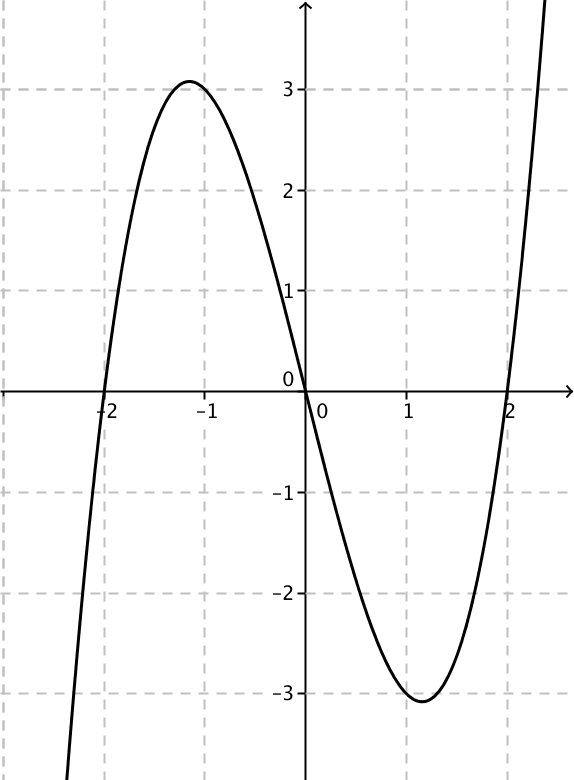
|  |  |  |
| --- | --- | --- |
| **Declare and real Let  Initialize as {} For all in the real numbers  For all in the real numbers  If then  Append to   else  Do NOT append to   Endif  Next  Next  Plot** |  | **For each value, the code Tests whether loops through all values. is a solution to .** |

**NESTED FOR-NEXT LOOPS:** Code like this is often called a “nested for-next loop” because the “For all …Next ” loop runs through all numbers each time a new value is chosen. One can imagine the outer “For all …Next ” loop as stepping through all numbers from to on the -axis, and for each choice of a number , one imagines the “For all …Next ” code as stepping through each number from to on the vertical number line that passes through the point . The picture below shows an example of this process (for a different function than in the example) at the step in the thought code where is 4:



In this way the “thought code” above checks every single point in the Cartesian coordinate plane to see if it is a solution to the equation .

* 1. Plot on the Cartesian plane (the figure drawn is called the graph of ).

Solution

Walk around and help students construct the graph in groups of two or individually by generating a table of order pairs, plotting those points, and then “connect-the-dots” between the points with a smooth curve. Remind students that they can use their work from Exercise 1 to get started. Note: While they do not have to generate the table in the same way that the thought code builds the set , once students have generated a table, ask them what order the thought code would have found the points in their table.

* 1. Describe how the "thought code” is similar to the set-builder notation .

Answer: Both generate sets by checking every point in the Cartesian plane, searching for points for which the equation is true.

* Go over this answer with your students. Ask for:  
    
  Domain of : all real numbers.

Range of : all real numbers.

The thought code describes a systematic way of imagining how all of the points in the graph of are found. However, the set should be thought of as the *end* *result* of the thought code; that is, the set contains all the points of the graph of . In this way, we can think of the set as the actual figure in the plane. (Students have already been introduced to this type of thinking in 8th grade using the simple example of associating a line as a geometric figure to the graph of a solution set of a linear equation. We will need this type of thinking in later grades to describe parabolas as graphs of quadratic functions, for example.)

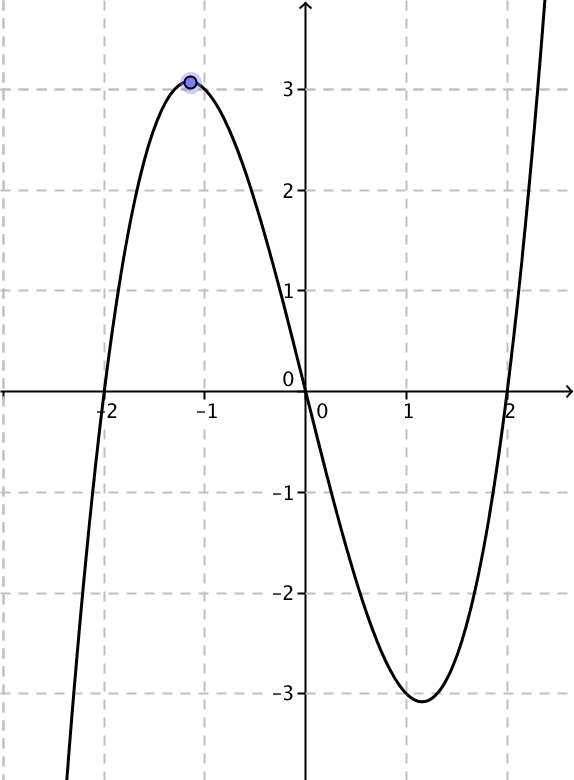
**IMPORTANT: Why are the graph of and the graph of the same set?** Note that the graph of was generated in Lesson 11 using different thought code than the code used to generate the graph of . Why do they generate the same set, then? Here’s why: For any given , the thought code for the graph of ranges over all values, which can be thought of as drawing a vertical line that passes through the point —see the picture of the vertical line at in the “nested for-next loop” discussion above. Because the definition of a function implies that each number in the domain is matched to one and only one value, , in the range, there can only be one point on that vertical line that satisfies the equation , namely the point generated by the thought code for the graph of . Since the graph of and the graph of the equation are both restricted to just the -values in the domain of , the two sets must be the same.

In the discussion above, the graph of intersects that vertical line through in exactly one point, , for in the domain of . This fact shows that the graph of any function satisfies the “vertical line test.” Do not introduce the “vertical line test” now. However, you can and should describe how the thought code for the graph of checks each point on the vertical line for a solution, and why, because of the definition of function, the code will find only one solution on that line for each in the domain of .

Note to teacher: Parts (c) and (d) of the exercise offer an excellent opportunity to check on your students’ 8th grade skills with square roots (standard 8.EE.2). However, if you feel they may need an easier problem so that students can scaffold up to parts c and d, you can ask students to answer parts c-f first using the function given by .

* 1. A *relative maximum* for the function occurs at the -coordinate of . Substitute this point into the equation to check that it is a solution to , and then plot the point on your graph.

Answer: Check that is a true number sentence. Divide both sides by multiply both sides by , and evaluate the square to get . Distributing the shows that is a solution.



You may need to remind students that . If time permits, allow them to use their calculators to see that the point really is the point they plotted on their graph.

* 1. A *relative minimum* for the function occurs at the -coordinate of . A similar calculation as you did above shows that this point is also a solution to . Plot this point on your graph.

Answer: Students should plot the point on their graphs approximately at .

* 1. Look at your graph. On what interval(s) is the function decreasing?

Answer: or .

* 1. Look at your graph. On what interval(s) is the function increasing?

Answer: or or .

Closing (2 minutes)

Lesson Summary

* **Graph of . Given a function whose domain and the range are subsets of the real numbers, the *graph of*  is the set of ordered pairs in the Cartesian plane given by**

**When we write for the graph of , it is understood that the domain is the largest set of real numbers for which the function is defined.**

* **The graph of is the same as the graph of the equation .**
* **Increasing/Decreasing. Given a function whose domain and range are subsets of the real numbers and is an interval contained within the domain, the function is called *increasing on the interval* if**

**whenever in .**

**It is called *decreasing on the interval if***

**whenever in .**

Exit Ticket (8 minutes)

Name \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Date\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Lesson 12: The Graph of the Equation**

**Exit Ticket**

* + - 1. Perform the instructions in the following programming code as if *you were a computer and your paper was the computer screen*:

|  |
| --- |
| **Declare integer For all from 2 to 7  If then  Print True  else  Print False  Endif Next** |

2. Let for in the domain

* + - * 1. Write out in words the meaning of the set notation:
        2. Sketch the graph of .



**Exit Ticket Sample Solutions**

1. Perform the instructions in the following programming code as if *you were a computer and your paper was the computer screen*:

|  |
| --- |
| **Declare integer For all from 2 to 7  If then  Print True  else  Print False  Endif Next** |

Answer: False, False, False, True, False, False.

1. Let for in the domain
   1. Write out in words the meaning of the set notation:

Answer: The set of all points in the Cartesian plane such that is between 0 and 2 inclusively and is true.

* 1. Sketch the graph of .



Problem Set Sample Solutions

1. Perform the instructions in the following programming code as if *you were a computer and your paper was the computer screen*:

|  |
| --- |
| **Declare integer For all from 1 to 6  If then  Print True  else  Print False  Endif Next** |

Answer: False, False, True, False, False, False.

1. Answer the following questions about the computer programming code:

|  |
| --- |
| **Declare integer Initialize as {} For all from 3 to 3  If then  Append to   else  Do NOT append to   Endif Next  Print** |

* 1. Perform the instructions in the programming code as if you were a computer and your paper was the computer screen:

Answer: .

* 1. Write a description of the set using set-builder notation.

Answer: .

1. Answer the following questions about the computer programming code:

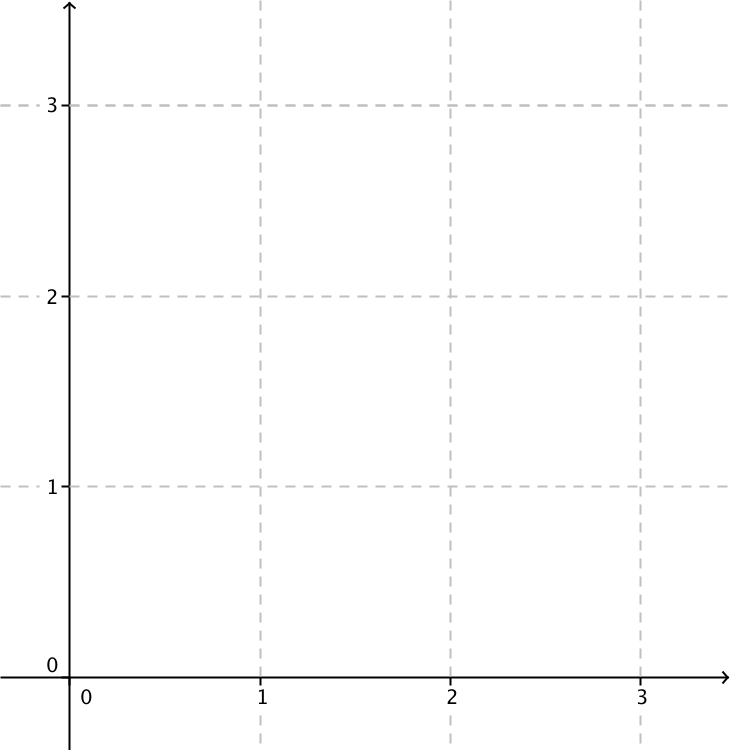
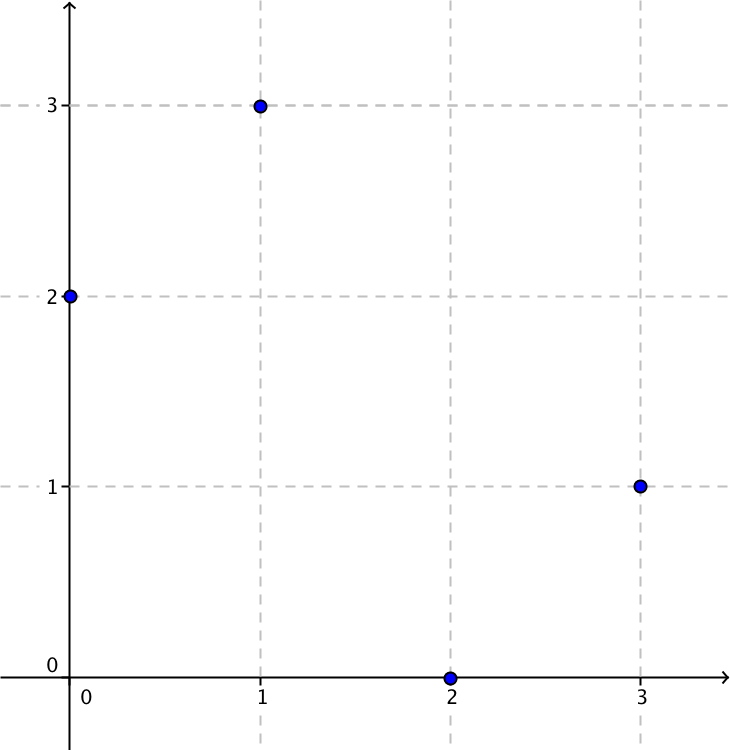
|  |
| --- |
| **Declare and integers Initialize as {} For all in   For all in   If then  Append to   else  Do NOT append to   Endif  Next  Next  Plot** |

* 1. Use the table below to record the decisions a computer would make when following the program instructions above. Fill in each cell with “Yes” or “No” depending on whether the ordered pair would be appended or not.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  |  |  |
|  | No | Yes | No | No |
|  | Yes | No | No | No |
|  | No | No | No | Yes |
|  | No | No | Yes | No |

* 1. Plot the set in the Cartesian plane.

Solution

1. Answer the following questions about the “thought code”:

|  |
| --- |
| **Declare and real Let  Initialize as {} For all in the real numbers  For all in the real numbers  If then  Append to   else  Do NOT append to   Endif  Next  Next  Plot** |

* 1. What is the domain of the function ?

Answer: all real numbers.

* 1. What is the range of the function

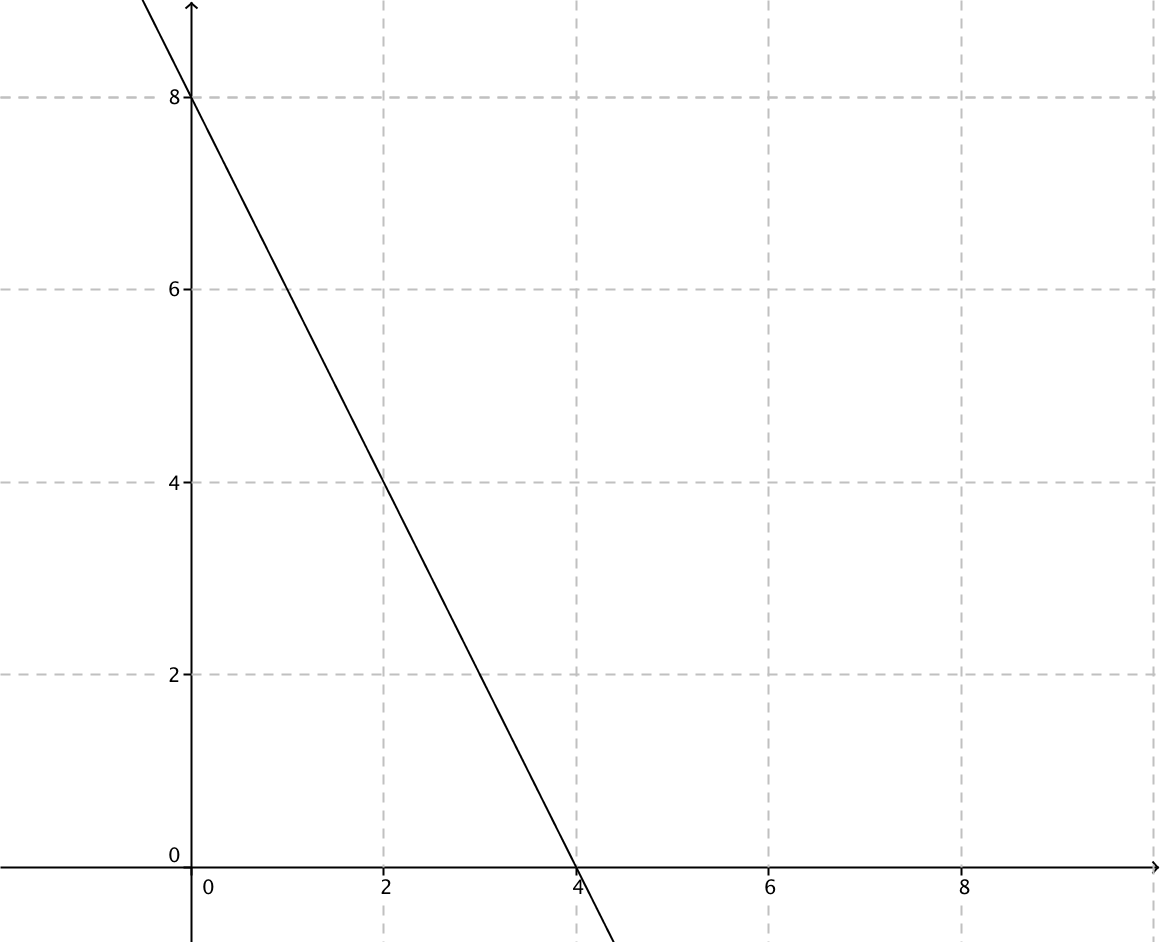
Answer: all real numbers.

* 1. Write the set generated by the “thought code” in set-builder notation.

Answer: .

* 1. Plot the set to obtain the graph of the function .

Answer:

**

* 1. The function is clearly a decreasing function on the domain of the real numbers. Show that the function satisfies the definition of decreasing for the points and on the number line, i.e., show that since , then .

Answer: and . Since , .

1. Sketch the graph of the functions defined by the following formulas and write the graph of as a set using set-builder notation. (Hint: For each function below you can assume the domain is all real numbers.)
2. Answer the following questions about the set:
   1. The equation can be rewritten in the form where . What are the domain and range of the function specified by the set?
      1. Domain:

**Answer: .**

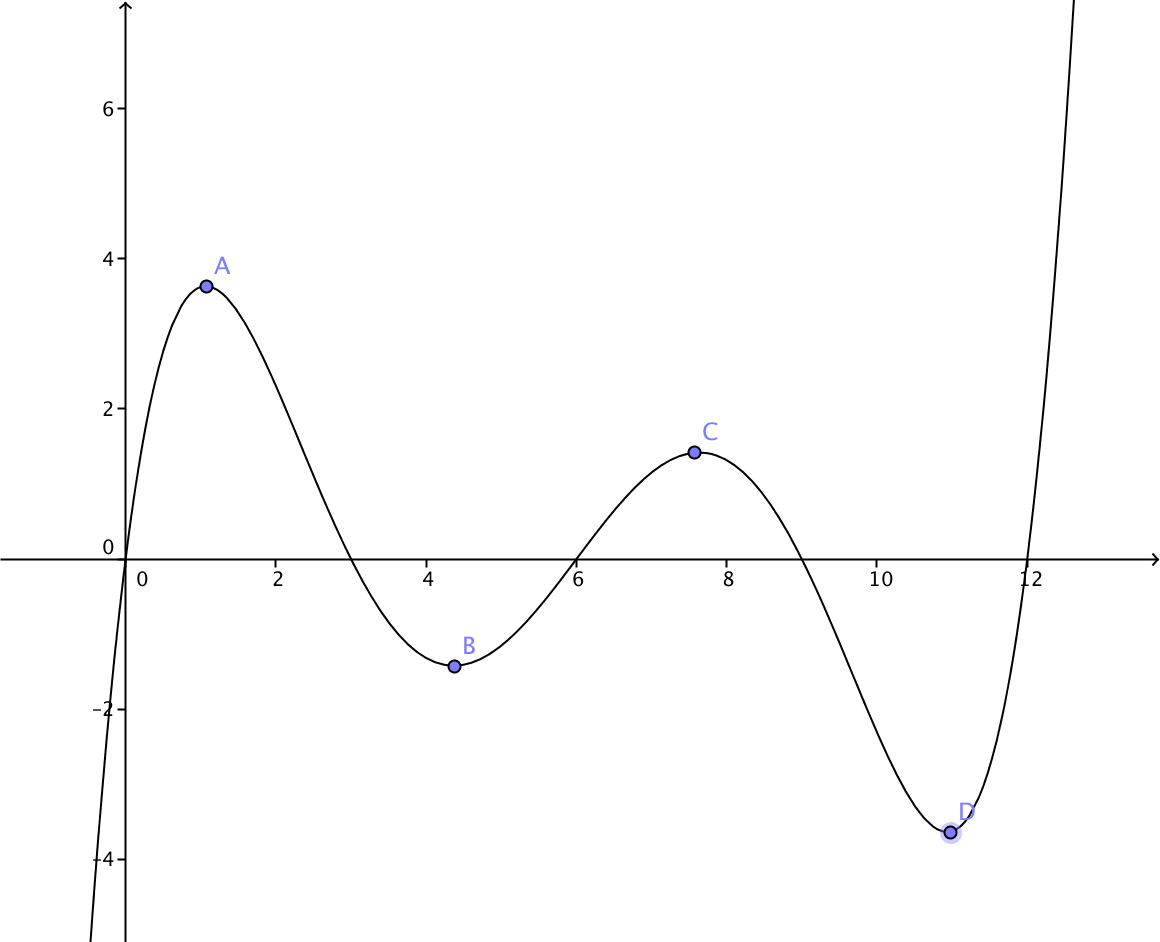
* + 1. Range:

**Answer: or for in the domain.**

* 1. Write “thought code” like in Exercise 4 that will generate and then plot the set.

Answer:

|  |
| --- |
| Declare and real Let  Initialize as {} For all such that   For all such that   If then  Append to   else  Do NOT append to   Endif  Next  Next  Plot |

1. Answer the following about the graph of a function below:   
     
     
   1. Which points (A, B, C, or D) are relative maximums?

Answer: A and C.

* 1. Which points (A, B, C, or D) are relative minimums?

Answer: B and D.

* 1. Name any interval where the function is increasing.

Answers will vary. Example: .

* 1. Name any interval where the function is decreasing.

Answers will vary. Example: